NODE FEATURES ADJUSTED STOCHASTIC BLOCK MODEL

Yun Zhang[1], Kehui Chen[1], Allan Sampson[1]
Kai Hwang[4] and Beatriz Luna[2,3]

[1]Dept. of Statistics, [2]Dept. of Psychiatry, [3]Center for the Neural Basis of Cognition,
University of Pittsburgh, Pittsburgh, PA.
[4]Helen Wills Neuroscience Institute, UC Berkeley, Berkeley, CA
June, 2018

ABSTRACT

Stochastic block model (SBM) and its variants are popular models used in community detection for network data. In this paper, we propose a feature adjusted stochastic block model (FASBM) to capture the impact of node features on the network links as well as to detect the residual community structure beyond that explained by the node features. The proposed model can accommodate multiple node features and estimate the form of feature impacts from the data. Moreover, unlike many existing algorithms that are limited to binary-valued interactions, the proposed FASBM model and inference approaches are easily applied to relational data that generates from any exponential family distribution. We illustrate the methods on simulated networks and on two real world networks: a brain network and an US air-transportation network.

KEY WORDS: stochastic block model, community detection, node features, air-transportation network, brain functional connectivity study.

Kehui Chen's effort is partially supported by NSF 1612458. The authors gratefully acknowledge Dr. Michael Hallquist for helpful discussions on the brain network data example. The authors thank the AE and referees for very helpful comments.

1 Introduction

In recent years, there has been increasing interest in statistical methodologies designed for network data. Network data takes the form of observed edges between nodes. Examples include brain networks (in which the nodes are segregated brain regions and edges are characterizations of white matter structural connectivity or brain's functional interactions) and social networks (in which the nodes are people and the edges may represent social interaction such as friendship or collaboration). The nodes and edges together define a network, often represented by an adjacency matrix, indicating the pairwise connection between nodes.

Community detection is a popular problem in network analysis. It has been a useful tool in identifying the important structures of many complex systems. Loosely speaking, network community refers to a subset of nodes that have similar profiles of connection to other nodes. Two classes of methods are commonly used for community detection. The first class of methods seeks community structure by optimizing a criterion that represents the quality of the partition of the network. These criteria come from a sense of what network communities should look like, lacking the interpretation of the data process that gives rise to the network. Though not orginated from a model, some popular methods such as simple spectral clustering (Von Luxburg, 2007) and Newman-Girvan Modularity (Newman and Girvan, 2004) has been proved to produce consistent estimation under stochastic block models (Bickel and Chen, 2009; Rohe et al., 2011; Lei et al., 2014). The second class of methods involves fitting a probabilistic model that has well-defined communities, where community detection is achieved by optimizing some statistical criterion linked to the assumed model, for example, using the likelihood. One of the most popular models is the stochastic block model (SBM) (Holland et al., 1983; Snijders and Nowicki, 1997; Nowicki and Snijders, 2001). The important assumptions of the SBM model are that each node belongs to one of the multiple blocks and the probability that an edge appears between any two nodes only depends on the memberships of the two nodes. Karrer and Newman (2011) proposed the degree corrected stochastic block model (DCBM) that allows degree inhomogeneity within blocks. Another popular model that shares the same goal of inferring node cluster labels is proposed in Handcock et al. (2007), where they extend the original latent space model proposed in Hoff et al. (2002) by combining a clustering model in the form of a mixture of Gaussians in the latent space

so that inference on cluster labels is attainable along with the latent positions. For a survey of statistical models used in analysing network data, see Goldenberg et al. (2010) and Kolaczyk (2009).

Despite the extensive literature on community detection, most of the proposed methods only focus on the observed edges of the network without taking into account the additional information of node features (or node attributes). In many networks, the similarities and distinctions in the node features have considerable impact on the pattern of linking. The nodes in different communities are commonly assumed to have distinct connectivity patterns while the impact of node features is usually in a more continuous fashion. For example, in the global airline network, there are more connections between large airports, and, in the social network, individuals who are more similar to one another in age and education are more likely to have interconnections (McPherson et al., 2001). It is generally expected that integrating node features and network topology can help us understand the network structure better than using the adjacency matrix alone or node features information alone.

The primary focus of this paper is to take node features into account in network analysis in order to capture the impact of node features on the network links, as well as to detect the residual community structure beyond that explained by the node features. For instance, in the brain connectivity study, all the nodes are naturally embedded in a three-dimensional brain space. Connectivity between adjacent nodes is sometimes over-represented due to technical reasons (Stanley et al., 2013). One needs to account for the spurious connectivity in adjacent nodes by removing the effect of spatial location so as to recover functionally distinct brain regions ("communities").

There are some recent attempts in integrating node features and network topology (Viennet et al., 2012; Liu et al., 2014; Yang et al., 2013; Binkiewicz et al., 2017; Zhang et al., 2016; Newman and Clauset, 2016). These efforts have provided great motivation for combining node features with community detection. In particular Binkiewicz et al. (2017) introduced a covariate assisted spectral clustering that leverages both node covariates and the graph in spectral techniques. Zhang et al. (2016) proposed to include edge weights as a function of node features to an analogue of modularity so that nodes having more similarity are more likely to be grouped into the same community. Newman and Clauset (2016) illustrates how to use or ignore the node feature data

depending on whether they contain useful information. However, the methods are mainly approaches aiming at improving community detection using node features that are aligned to the communities to a certain degree, while we take a different perspective. We build a generative stochastic model that models the node features effect and the community effect on the network additively. The impact of node features on the probability of linking between two nodes is usually a function of the similarities in the two node features. The proposed method can assist community detection in the sense that it estimates and accounts for the effect of covariates, and so as to reveal the hidden community structure.

The model we propose is a feature adjusted stochastic block model (FASBM), which combines a block model component with community structures and a single-index function to incorporate node features. As a generative model, the FASBM model assumes that the connectivity probability between two nodes i and j is determined by their communities, and also a smooth function of the node features. The heterogeneity within a block is explained by the continuous effect of specific node features. The estimation of the FASBM model involves discovering the optimal block partition as part of the model estimation while capturing the impact of node features on the network links. The proposed model builds upon the stochastic block model (SBM) and, thus, inherits the merits of block models. With a semi-parametric single-index component, it is also adequately flexible to accommodate multiple node features with no prior information about the contribution of the features. Moreover, unlike many existing algorithms that are limited to binary-valued interactions, the proposed FASBM model and estimation approaches are applicable to relational data that are generated from any exponential family distribution and are not restricted to being only Bernoulli.

The rest of the paper is organized as follows. We describe the proposed feature adjusted stochastic block model (FASBM) in Section 2. Section 3 is devoted to a detailed description of the estimation procedures. The performance of the proposed method is demonstrated on a range of simulated networks in Section 4 and in Section 5 is applied to a functional brain network and an US air-transportation network . The paper is concluded with a short discussion in Section 6.

2 Feature Adjusted Stochastic Block Model (FASBM)

2.1 Stochastic Block Model and Likelihood Inference

We consider undirected networks in our paper, and self-loops are not allowed unless otherwise specified. Most of the networks that have been studied are binary in nature, that is, the edges between nodes indicate the presence or absence of an interaction. Binary network can be represented by a binary adjacency matrix $Y = (Y_{ij})_{1\leq i,j\leq m}$, where $Y_{ij} = 1$ if there is an edge between node i and a different node j, and $Y_{ij} = 0$ otherwise.

The SBM has been developed in concordance with the notion of structural equivalence in a graph. Let K be the number of non-overlapping communities, m be the number of nodes and $\boldsymbol{r}$ be a vector of community labels with $r_i = k$ if node $i, i = 1, \cdots, m$, belongs to the community k , $k = 1, \cdots, K$. For the SBM, the adjacency matrix Y is generated by

$$Y_{ij} = \begin{cases} \text{independent Bernoulli with probability } \mu_{ij} & \text{if} \quad i < j \\ 0 & \text{if} \quad i = j \\ Y_{ji} & \text{if} \quad i > j \end{cases} \tag{1}$$

A stochastic block model is parameterized by a pair of $(\boldsymbol{r}, B)$, where B is a $K \times K$ symmetric matrix,

$$E(Y_{ij}) = \mu_{ij} = B_{r_i r_j}. \tag{2}$$

Under the SBM, each node belongs to one of the multiple blocks, and the probability that an edge appears between any two nodes only depends on the block memberships of the two nodes. The primary interest of community detection is concerned with estimating $\boldsymbol{r}$. It has been studied in Bickel and Chen (2009); Choi et al. (2012); Zhao et al. (2012); Celisse et al. (2012) that blockmodels and the corresponding likelihood-based algorithms are (asymptotically) unbiased and lead to the detection of the correct community structure. Let $L(Y; B, \boldsymbol{r})$ denote the log-likelihood function $L(Y; B, \boldsymbol{r}) = \sum_{i=1}^{m} \sum_{j=i+1}^{m} \{Y_{ij} \log(B_{r_i r_j}) + (1 - Y_{ij}) \log(1 - B_{r_i r_j})\}$. Finding the global maximum involves maximizing the likelihood function over all possible label assignments, which is computationally infeasible. Some types of greedy label-switching algorithms for maximizing the likelihood function have been proposed and work well in practice. Another

popular community detection algorithm, simple spectral clustering (Von Luxburg, 2007), has also been proved to be consistent under SBM (Rohe et al., 2011; Lei et al., 2014). Bickel and Chen (2009) also proved that under some conditions, partitions obtained from the Newman-Girvan Modularity (Newman and Girvan, 2004) are consistent estimators of block partitions under the SBM, although the algorithm itself is not based on generative models.

2.2 Generalized semi-parametric single-index model

We aim to find a way to incorporate feature information into the stochastic block model, and meanwhile account for three practical considerations:

1) There may be multiple node features influencing the connection probability.

2) In the general case, we may not have good knowledge of how node features impact connections.

3) Many networks have relational data indicating differing strengths of interactions. For example, in a brain network there may be stronger or weaker reactions between two regions of interest, or in a collaborative research network there may be more or fewer co-authored papers between two researchers. Dichotomizing the strength of interaction would clearly destroy potentially valuable information.

We propose the Feature Adjusted Stochastic Block Model (FASBM) as follows. The network Y is generated by

$$Y_{ij} = \begin{cases} \text{independent exponential family with mean } \mu_{ij} & \text{if} \quad i < j \\ 0 & \text{if} \quad i = j \\ Y_{ji} & \text{if} \quad i > j \end{cases} \tag{3}$$

The distributions we consider here are mainly in one-parameter exponential family (uniquely determined by μ_{ij}). We allow for an unknown scaling parameter such as the variance in normal distribution. Our algorithm does not estimate the nuisance scaling parameter. Further specification of the mean μ_{ij} is as follows

$$E(Y_{ij}) = \mu_{ij} = g^{-1}(\boldsymbol{\theta}_{r_i r_j} + f(\boldsymbol{\beta}^T \boldsymbol{z}_{ij})), \quad \text{with } \|\boldsymbol{\beta}\| = 1. \tag{4}$$

where g is a known link function, $\boldsymbol{\theta}$ is a $K \times K$ symmetric matrix that captures the block-wise effect, f is an unknown smooth function that will be estimated non-parametrically, $\boldsymbol{z}_{ij}$ is a p-dimensional vector of covariates and $\boldsymbol{\beta}$ is the p-dimensional linear coefficient. Here $\boldsymbol{z_{ij}}$ is selected in a manner depending on the node features $\boldsymbol{f}_i$ and $\boldsymbol{f}_j$ and we assume $\boldsymbol{z_{ij}} = \boldsymbol{z_{ji}}$. Suppose that in a brain network, we are interested in assessing the impact of spatial locations on brain connectivity, the physical distance between two brain regions may be a sensible choice, i.e., $\boldsymbol{z}_{ij} = d(\boldsymbol{f}_i, \boldsymbol{f}_j)$ where d is a distance measure and $\boldsymbol{f}_i$ is the location of region i in the three-dimensional brain space. Noting that we basically model the probability of the presence of an edge as two parts, one is a discrete part that captured by the $\boldsymbol{\theta}$ matrix, and the other part captured by the smooth function f. For model identifiability, we require $\beta^T \boldsymbol{z}_{ij}$ to take values on an interval and none of the covariates is perfectly aligned with the communities. In the extreme case that we have two communities, and the covariate takes one value in community 1 and takes another value in community 2, which means the covariates are completely aligned with the community, we get into an identifiability problem. Note that the covariates can still be correlated with the communities, such as settings in our simulation II. Our model encompasses many types of relational data generated from an exponential family distribution. Families that generate the well known class of generalized linear models are all extendable in the same way to the FASBM. The component $f(\boldsymbol{\beta}^T \boldsymbol{z_{ij}})$ can be referred to as a single-index component (Carroll et al., 1997). For identifiability and for easier interpretation, the restriction $\|\boldsymbol{\beta}\| = 1$ is used with the first component of $\boldsymbol{\beta}$ being positive, and we also set $f(x_0) = 0$ for a chosen constant x_0. Single-index models have been proven to be an efficient way to avoid fitting multivariate nonparametric regression functions.

The proposed FASBM can be viewed as a generalized semi-parametric single index model (4), which consists of two parts: i) block model parameter $\boldsymbol{\theta}$ that enters the model as a parametric component, retaining the generality and tractability of the block model and ii) a single-index component $f(\boldsymbol{\beta}^T \boldsymbol{z_{ij}})$. The non-parametric function f is flexible to characterize nonlinear covariate effects, while $\boldsymbol{\beta}^T \boldsymbol{z}_{ij}$ reduces the dimension of the covariates. When no feature is concerned or covariates have no effect on node connections, FASBM becomes a generalization of the stochastic block model to accommodate relational data drawn from exponential families other than Bernoulli distribution. The classic SBM is obviously a special case of FASBM.

3 Maximum likelihood estimation for FASBM

In this section, we introduce the fitting algorithms for our proposed model.

Consider $m(m-1)/2$ independent random variables Y_{ij} from exponential family distribution. The log-likelihood function in the canonical form with a canonical link is given as

$$\begin{aligned} L(Y;\boldsymbol{\theta},\boldsymbol{r},\boldsymbol{\beta}) &= \sum_{i=1}^{m}\sum_{j=i+1}^{m}\{(Y_{ij}\gamma_{ij}-b(\gamma_{ij}))/\phi + a(y_{ij},\phi)\} \text{ and} \\ \gamma_{ij} &= \boldsymbol{\theta}_{r_i r_j} + f(\boldsymbol{\beta}^T \boldsymbol{z}_{ij}). \end{aligned} \tag{5}$$

Here ϕ is a nuisance parameter, and functions $b(\cdot), a(\cdot,\cdot)$ are completely determined by the log-likelihood function of the data. In the case that Y_{ij} is binary data assumed to follow a Bernoulli distribution, the canonical link function g is the logit function, $b(\gamma) = \log(1+\exp(\gamma))$, and $a(y) \equiv 1$. Our goal is to maximize the logarithm of the likelihood function with respect to the unknown model parameters $\boldsymbol{\theta}, \boldsymbol{\beta}, f$, along with the node label assignment vector $\boldsymbol{r}$. Because an exact maximization of the (5) is computationally intractable, we propose an approach that alternates between two stages of maximization: first with respect to the parameters in the block model component, $\boldsymbol{r}$ and $\boldsymbol{\theta}$, and then with respect to the parameters in the single-index model component, f and $\boldsymbol{\beta}$. We adapt the likelihood-based algorithms for the SBM to stage 1 and the estimation procedures for fitting single-index models (Carroll et al., 1997) to stage 2. Note that we used the canonical link function to explicitly write equation (5). In fact, the algorithm works for general link functions. Detailed descriptions of the algorithms are provided in the Subsection 3.2, and the code is publicly available on authors webpage.

In light of the fact that it has been proved in Bickel and Chen (2009) that partitions with likelihood-based algorithms for the SBM are consistent, we would expect a good chance of recovering membership consistently, as long as we can consistently estimate the single-index part f and $\boldsymbol{\beta}$. On the other hand, given $\boldsymbol{r}$, our model can be viewed as a generalized semi-parametric single-index model and consistency of the estimates $f, \boldsymbol{\beta}$ and $\boldsymbol{\theta}$ follows from Carroll et al. (1997). Empirically we show satisfactory performance of the algorithm as detailed in the Section 4.

3.1 Preliminaries

Local polynomial maximum likelihood estimation: We estimate f using local polynomial maximum likelihood estimation. Imagine for a moment that node membership $\boldsymbol{r}$, and $\boldsymbol{\theta}$, $\boldsymbol{\beta}$ are fixed. We estimate the function f for each point x_0 by maximizing the local kernel-weighted log-likelihood

$$\begin{aligned} L(Y;\boldsymbol{\theta},\boldsymbol{r},\boldsymbol{\beta}) &= \sum_{i=1}^{m}\sum_{j=i+1}^{m}\{(Y_{ij}\gamma_{ij}-b(\gamma_{ij}))/\phi+a(y_{ij},\phi)\}K_h(\boldsymbol{\beta}^T\boldsymbol{z}_{ij}-x_0) \text{ and} \\ \gamma_{ij} &= \boldsymbol{\theta}_{r_ir_j}+b_0+b_1(\boldsymbol{\beta}^T\boldsymbol{z}_{ij}-x_0))+\cdots+b_p(\boldsymbol{\beta}^T\boldsymbol{z}_{ij}-x_0)^p. \end{aligned} \tag{6}$$

with respect to $(b_0, b_1, \ldots, b_p)$ and then $\hat{f}(x_0) = \hat{b}_0$ and $\hat{f}^{(1)}(x_0) = \hat{b}_1$. Here $f(x)$ is locally approximated by a polynomial function near x_0:

$$f(x) \approx f(x_0) + f^{(1)}(x_0)(x-x_0) + \cdots + \frac{1}{p!}f^{(p)}(x-x_0)^p \equiv b_0 + b_1(x-x_0) + \cdots + b_p(x-x_0)^p,$$

and $K_h(\cdot) = K(\cdot/h)/h$ is a rescaled kernel function $K(\cdot)$ with bandwidth h, which places more weight on those observations closer to x_0. In general, the finite sample performance is not very sensitive to the choice of p within a reasonable range. Previous work (Fan and Gijbels, 1996) recommend to choose the degree of polynomial p as the desired derivative plus one, i.e., use local linear approximation for estimating f, and use local quadratic approximation for estimating the first derivative $f^{(1)}$. We used $p = 2$ in our simulations since we also need $f^{(1)}$ in the step of updating $\boldsymbol{\theta}$.

Fisher Scoring algorithm: Our estimation of $\boldsymbol{\theta}$ and $f(\cdot)$, $\boldsymbol{\beta}$ all use the Fisher Scoring algorithm for maximum likelihood estimation. Consider a random variable y with a distribution in the exponential family. The log-likelihood for one observation can be expressed as $l(y,\gamma,\phi) = [(y\gamma - b(\gamma))/\phi + a(y,\phi)]$ for known functions $b(\cdot), a(\cdot,\cdot)$, and it is easy to show that $E(y) = \mu = b'(\gamma)$ and $\mathrm{Var}(y) = b''(\gamma)\phi = V(\mu)$. When alternating between the estimation of $\boldsymbol{\theta}$, $f(\cdot)$ and $\boldsymbol{\beta}$, the proposed model $\mu = g^{-1}(\theta + f(\boldsymbol{\beta}^T\boldsymbol{z}))$ can be written as $g(\mu) = \eta(\boldsymbol{B})$, with its respective form of η and unknown parameters $\boldsymbol{B}$. By the chain rule and properties of exponential family,

score function $U(\boldsymbol{B})$ for N observations becomes

$$\begin{aligned} U(\boldsymbol{B}) = \sum_{s=1}^{N} \boldsymbol{u}_s = \sum_{s=1}^{N} \frac{\partial l_s(\boldsymbol{B})}{\partial \boldsymbol{B}} &= \sum_{s=1}^{N} \frac{\partial l_s}{\partial \gamma_s} \frac{\partial \gamma_s}{\partial \mu_s} \frac{\partial \mu_s}{\partial \eta_s} \frac{\partial \eta_s}{\partial \boldsymbol{B}} w_s \\ &= \sum_{s=1}^{N} \frac{y_s - \mu_s}{\phi} \frac{1}{V(\mu_s)} g^{-1\prime}(\eta_s) \frac{\partial \eta_s}{\partial \boldsymbol{B}} w_s \\ &= \frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{B}} \boldsymbol{W_1} \boldsymbol{W} (\boldsymbol{y} - \boldsymbol{\mu}) \end{aligned}$$

with diagonal matrix $[\boldsymbol{W_1}]_{ss} = \dfrac{g^{-1\prime}(\eta_s)}{\phi V(\mu_s)}$ and diagonal weight matrix $[\boldsymbol{W}]_{ss} = w_s$. The weight matrix $\boldsymbol{W}$ is simply an identity matrix when maximizing the global log-likelihood for the estimation of $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$. When considering the local kernel-weighted log-likelihood for estimating $f(x_0)$, the local kernel-weight w_s for each observation is specified in Section 3.1.

The Hessian matrix and Information matrix become:

$$H(\boldsymbol{B}) = \frac{\partial U(\boldsymbol{B})}{\partial \boldsymbol{B}} = \sum_{s=1}^{N} (y_s - \mu_s) \frac{\partial \left([\boldsymbol{W_1}]_{ss} w_s \dfrac{\partial \eta_s}{\partial \boldsymbol{B}} \right)}{\partial \boldsymbol{B}} + [\boldsymbol{W_1}]_{ss} w_s \frac{\partial \eta_s}{\partial \boldsymbol{B}} \frac{\partial (y_s - \mu_s)}{\partial \boldsymbol{B}},$$

and

$$\begin{aligned} I(\boldsymbol{B}) = -E(H(\boldsymbol{B})) = \sum_{s=1}^{N} [\boldsymbol{W_1}]_{ss} w_s \frac{\partial \eta_s}{\partial \boldsymbol{B}} \frac{\partial \mu_s}{\partial \boldsymbol{B}} &= \sum_{s=1}^{N} [\boldsymbol{W_1}]_{ss} w_s \frac{\partial \eta_s}{\partial \boldsymbol{B}} \frac{\partial \mu_s}{\partial \eta_s} \frac{\partial \eta_s}{\partial \boldsymbol{B}} \\ &= \sum_{s=1}^{N} [\boldsymbol{W_1}]_{ss} w_s \frac{\partial \eta_s}{\partial \boldsymbol{B}} g^{-1\prime}(\eta_s) \frac{\partial \eta_s}{\partial \boldsymbol{B}} \\ &= \frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{B}} \boldsymbol{W_2} \boldsymbol{W} \left(\frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{B}} \right)^T \end{aligned}$$

with diagonal matrix $[\boldsymbol{W_2}]_{ss} = \dfrac{(g^{-1\prime}(\eta_s))^2}{\phi V(\mu_s)}$. Given $\boldsymbol{B}^{(l)}$ at the previous step, by the Fisher Scoring algorithm, the updated $\hat{\boldsymbol{B}}^{(l+1)} = \hat{\boldsymbol{B}}^{(l)} + I^{-1}(\hat{\boldsymbol{B}}^{(l)}) U(\hat{\boldsymbol{B}}^{(l)})$,

$$\hat{\boldsymbol{B}}^{(l+1)} = \hat{\boldsymbol{B}}^{(l)} + \left(\frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{B}} \boldsymbol{W_2} \boldsymbol{W} \left(\frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{B}} \right)^T \right)^{-1} \frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{B}} \boldsymbol{W_1} \boldsymbol{W} (\boldsymbol{y} - \boldsymbol{\mu}) \Big|_{(l)} \tag{7}$$

The approach for updating f, $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ all fall into the above framework with its respective η and unknown parameters $\boldsymbol{B}$, which will be specified in Section 3.2. The weight matrices $\boldsymbol{W}$ are the

kernel weights for local likelihood estimation, and only used in updating f. Given $\boldsymbol{\eta}$, the link function g, and the distribution of Y, matrices $\boldsymbol{W_1}$ and $\boldsymbol{W_2}$ can be computed according to the formula given above.

3.2 The Algorithm

The algorithm takes the relational data Y, covariates $\boldsymbol{z}$, assumed number of communities K and the assumed distribution for Y (within the exponential family) as input, and output all the estimated model components. Before we demonstrate the detailed algorithms, we convert the upper triangle (excluding the diagonal) of Y into a vector $Y^*_{N\times 1} = \left(Y_{12}, \cdots, Y_{(m-1)m}\right)^T$ where $N = m(m-1)/2$, and accordingly let $\boldsymbol{Z}^*_{N\times p} = \left(\boldsymbol{z}_{12}, \cdots, \boldsymbol{z}_{(m-1)m}\right)^T$. We use $Y^*_{s(ij)}$ and $\boldsymbol{z}^*_{s(ij)}$ for the correspondence between s and the pair (i, j) when necessary.

(a) Initialization: Let $\hat{f}(\cdot) = 0$, each entry of $\hat{\boldsymbol{\beta}} = \sqrt{1/p}$, assign initial labels $\boldsymbol{r}$ by k-means on the rows of Y matrix.

(b) Updating $\boldsymbol{\theta}$ and $\boldsymbol{r}$: Given $\hat{f}^{(o)}$ and $\hat{\boldsymbol{\beta}}^{(o)}$, obtain $\hat{\boldsymbol{\theta}}^{(o+1)}$ and $\hat{\boldsymbol{r}}^{(o+1)}$ by repeating steps of updating $\boldsymbol{\theta}$ and $\boldsymbol{r}$ iteratively until $\boldsymbol{r}$ is unchanged.

Suppressing the superscript (o), given the current $\hat{f}$ and $\hat{\boldsymbol{\beta}}$, each iteration of updating $\boldsymbol{\theta}$ and $\boldsymbol{r}$ involves two steps:

(i) Given $\hat{\boldsymbol{r}}^{(q-1)}$, update $\hat{\boldsymbol{\theta}}^{(q)}$ through (7) by reparameterizing the upper triangle of $\boldsymbol{\theta}_{K\times K}$ into $\boldsymbol{B}_{P\times 1} = (\boldsymbol{\theta}_{11}, \cdots, \boldsymbol{\theta}_{1K}, \boldsymbol{\theta}_{22}, \cdots, \boldsymbol{\theta}_{KK})^T$ with $P = K(K+1)/2$. Here $\eta_{s(ij)} = \boldsymbol{x}^T_{s(ij)}\boldsymbol{B} + f(\boldsymbol{\beta}^T\boldsymbol{z}^*_{s(ij)})$ for $s = 1, \ldots, N$, where $\boldsymbol{x}_{s(ij)}$ has only one 1 indicating the memberships (r_i, r_j), otherwise zero.

(ii) Given $\hat{\boldsymbol{\theta}}^{(q)}$, the community label for ith node $r_i^{(q)}$ is updated by minimizing the negative log-likelihood through the greedy label-switching algorithm (Stephens, 2000) as follows:

$$\hat{r_i}^{(q)} = \arg\min_{k\in\{1,\cdots,K\}} \sum_{j=1}^{m} \{-Y_{ij}\log[g^{-1}(\hat{\boldsymbol{\theta}}^{(q)}_{k,r_j^{(q-1)}} + \hat{f}(\hat{\boldsymbol{\beta}}^T\boldsymbol{z_{ij}}))] - (1-Y_{ij})\log[1-g^{-1}(\hat{\boldsymbol{\theta}}^{(q)}_{k,r_j^{(q-1)}} + \hat{f}(\hat{\boldsymbol{\beta}}^T\boldsymbol{z_{ij}}))]\}.$$

(c) Updating $\boldsymbol{\beta}$ and f: Given $\hat{\boldsymbol{\theta}}^{(o+1)}$ and $\hat{\boldsymbol{r}}^{(o+1)}$, obtain $\hat{f}^{(o+1)}$ and $\hat{\boldsymbol{\beta}}^{(o+1)}$ by iterating between updating $\boldsymbol{\beta}$ and f until $\frac{\|\hat{f}^{(q)} - \hat{f}^{(q-1)}\|}{\|\hat{f}^{(q-1)}\|} \leq \epsilon$ for a suitably chosen small constant ϵ, where $\|\cdot\|$ denotes L_2 norm and q denotes the index of iteration consisting of updating $\boldsymbol{\beta}$ and $\boldsymbol{f}$.

Omitting the superscript (o), given the current $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{r}}$, each iteration of updating $\boldsymbol{f}$ and $\boldsymbol{\beta}$ involves two steps:

(i) Given $\hat{f}^{(q-1)}$, $\hat{\boldsymbol{\beta}}^{(q)}$ is obtained through (7) by viewing $\boldsymbol{B} = \boldsymbol{\beta}$. Here $\eta_{s(ij)} = \boldsymbol{\theta}_{r_i r_j} + f(\boldsymbol{B}^T \boldsymbol{z}^*_{s(ij)})$. Note that $\hat{\boldsymbol{\beta}}$ need to be normalized to meet $\|\boldsymbol{\beta}\| = 1$.

(ii) Given $\hat{\boldsymbol{\beta}}^{(q)}$, we fit $\hat{f}(\cdot)$ at a fixed but fine grid of points and subsequently using interpolation to get the other values. Take one of the grid points x_0 for example, $\hat{f}(x_0)$ is updated through (7) using the local likelihood approach. Here, $\boldsymbol{B} = (b_0, b_1, b_2)$, $\eta_{s(ij)} = \boldsymbol{\theta}_{r_i r_j} + b_0 + b_1(\boldsymbol{\beta}^T \boldsymbol{z}^*_{\boldsymbol{s(ij)}} - x_0)) + b_2(\boldsymbol{\beta}^T \boldsymbol{z}^*_{\boldsymbol{s(ij)}} - x_0))^2$, $[\boldsymbol{W}]_{ss} = K_h(\hat{\boldsymbol{\beta}} \boldsymbol{z}^*_{\boldsymbol{s}} - x_0)$, $\hat{f}(x_0) = \hat{b}_0$ and $\hat{f}^{(1)}(x_0) = \hat{b}_1$.

(d) Iterate between steps (b) and (c) until $\frac{\|\hat{f}^{(o+1)} - \hat{f}^{(o)}\|}{\|\hat{f}^{(o)}\|} \leq \epsilon$ for a suitably chosen small constant ϵ.

4 Simulation Studies

We conduct four simulations total. Simulation I and II are designed to investigate the performance of the proposed method under different types and levels of node influence, including cases where node covariates are generated with or without alignment to node communities, and edge probability depends on node features through function f with various levels of influence, including the case $f \equiv 0$. Simulation III is designed to investigate the performance of the proposed method when there are nuisance nodal covariates i.e., a subset of the covariates have no impact on the edge probability, but the overall f is not zero. Simulation IV is designed to further investigate the performance under various shapes of f functions.

In **Simulation I**, the network generation procedure takes the following steps: first, generate labels for m nodes independently with $P(r_i = 1) = \cdots = P(r_i = K) = 1/K$; second, generate node feature x and compute the l_2 distance between x_i and x_j, denoted by z_{ij}; finally, the edges

between node i and node j are generated independently as $Y_{ij} \sim \text{Bernoulli}\left(g^{-1}(\boldsymbol{\theta}_{r_i r_j} + f(z_{ij})\right)$, where g is the logit function. The values of $\boldsymbol{\theta}$ for $K = 2$ and $K = 3$ are as follows,

$$\boldsymbol{\theta} = \text{logit} \begin{pmatrix} 0.5 & 0.2 \\ 0.2 & 0.2 \end{pmatrix}$$

$$\text{and} \quad \boldsymbol{\theta} = \text{logit} \begin{pmatrix} 0.5 & 0.2 & 0.2 \\ 0.2 & 0.3 & 0.2 \\ 0.2 & 0.2 & 0.1 \end{pmatrix}.$$

We let $f = a\sin(-8z_{ij})$, with a taking different values, $0, 1.4$ or 1.8. The node covariates x are generated uniformly from interval (0, 1). In **Simulation II**, we keep the same settings as in simulation I, $K = 2$, but generate covariates x_i according to a mixture Gaussian distribution, where x_i is from $N(-1, 1)$ if node i is in community 1, otherwise x_i is from $N(1, 1)$. In this case, the nodal covariate values are aligned to the communities.

In both simulations, we compare the community detection results with the likelihood-based inference of SBM (SBML), the simple spectral clustering (SPEC), the joint community detection criterion proposed in Zhang et al. (2016), and the covariate-assisted spectral clustering (CASC) proposed in Binkiewicz et al. (2017). We consider two measures to quantify the performance in terms of the agreement between the true $\boldsymbol{r}$ and $\hat{\boldsymbol{r}}$. The first measure is the average misclassification rates (ERR), quantifying the overall proportion of mis-clustered nodes (Girvan and Newman, 2002). We also adopt the normalized mutual information criterion (NMI) (Kvalseth, 1987) to measure clustering quality, where higher values indicate better matching.

It is worth mentioning that for methods based on spectral clustering, including SPEC, JCDC and CASC, one needs to choose the dimension d of spectral embedding. In the simulation, we tried different d values for these three methods, and reported the one with the best performance. There are two additional tuning parameters, α and w_n, users need to specify, in using JCDC method. In our simulations, α is set to 1, which is the same setting used for all the simulations and data analysis reported in Zhang et al. (2016). We varied the value of w_n and picked the value that favors JCDC method the most in terms of normalized mutual information. Also all the three algorithms require the number of communities to be known in advance, and we used the

true K in the simulation. Determining the number of communities is gaining increasing interest recently (Chen and Lei, 2018; Bickel and Sarkar, 2015). In the Section 5, we used the network cross-validation (NCV) method proposed by Chen and Lei (2018) to determine the number of communities.

Table 1 - Table 3 show the results of 100 simulation runs for simulations I and II. In the situations where features have no impact on the network topology, i.e., $a = 0$, FASBML and SBML perform equally well followed by SPEC and CASC. All the methods perform better as m increases as, with more links, there is effectively more data to use for fitting the model. The performance of other methods deteriorates rapidly as the amount of f effect increases. On the other hand, the partition found by FASBML still has very good agreement with the actual partition in the presence of large feature influence, and the performance improves as m increases. The inferiority of SBM relative to FASBM in these scenarios is understandable as FASBM always uses both the network topology and the features whereas SBM completely ignores feature influence. In addition, the fact that FASBML and SBML have equally good performance when $a = 0$ confirms the robustness of FASBML to the case where all covariates are irrelevant. The performance of JCDC and CASC relies on the relationships between the covariate and the communities. In their settings, the edge probability does not directly depend on covariates after accounting the community structure (though the covariates are aligned with communities), and it corresponds to $f \equiv 0$ in our setting. We do not mean to claim superiority by the comparison here, as these methods take somewhat different perspectives and have different applications.

In **simulation III**, in addition to having a covariate x_1 generated uniformly from interval $(0, 1)$, we generate covariate x_2 according to a mixture Gaussian distribution: in the case of $K = 2$, x_{2i} is from $N(-1, 1)$ if node i is in community 1, otherwise x_i is from $N(1, 1)$; in the case of $K = 3$, x_{2i} is from $N(-2, 1)$ if node i is in community 1, from $N(0, 1)$ if node i is in community 2, otherwise x_{2i} is from $N(2, 1)$. Let $f_1 = 1.8 \sin(-\frac{4}{3}(0.2z_{1ij} + 0.9798z_{2ij}))$, and $f_2 = 1.8 \sin(-\frac{4}{3}(z_{2ij}))$, where z_{1ij} and z_{2ij} are the L_2 distance between x_{1i} and x_{1j}, and between x_{2i} and x_{2j}, respectively. In the first case, $\beta_1 = 0.2$ and $\beta_2 = 0.9798$, where the β vector is set to have norm equal 1. In the second case, $\beta_1 = 0$ and $\beta_2 = 1$, which implies that x_1 is a nuisance covariate. Fitting FASBM using two covariates, we found that the estimated coefficients

Table 1: Results of simulation I, $K = 2$. The average misclassification rates (ERR) and normalized mutual information (NMI) are shown together with their standard deviations enclosed in parentheses for varying a in $f = a\sin(-8z_{ij})$, and varying number of nodes m. Numbers in bold indicate the best performance.

		$m = 100$					$m = 200$					$m = 400$				
		FASBML	SBML	SPEC	JCDC	CASC	FASBML	SBML	SPEC	JCDC	CASC	FASBML	SBML	SPEC	JCDC	CASC
$a = 0$	ERR	**0.012**	**0.012**	0.041	0.240	0.021	**0.0004**	**0.0004**	0.006	0.231	0.002	**0**	**0**	0.0003	0.227	**0**
		(0.010)	(0.010)	(0.024)	(0.04)	(0.016)	(0.0015)	(0.0015)	(0.006)	(0.024)	(0.003)	(0)	(0)	(0.0009)	(0.019)	(0)
	NMI	**0.924**	**0.924**	0.783	0.335	0.872	**0.997**	**0.997**	0.955	0.371	0.988	**1**	**1**	0.998	0.377	**1**
		(0.060)	(0.060)	(0.100)	(0.063)	(0.083)	(0.013)	(0.013)	(0.040)	(0.035)	(0.022)	(0)	(0)	(0.008)	(0.023)	(0)
$a = 1.4$	ERR	0.157	0.443	0.128	0.303	**0.093**	**0.012**	0.469	0.079	0.286	0.045	**0.0001**	0.481	0.045	0.354	0.021
		(0.200)	(0.087)	(0.041)	(0.078)	(0.040)	(0.067)	(0.024)	(0.031)	(0.082)	(0.025)	(0.0004)	(0.014)	(0.020)	(0.119)	(0.013)
	NMI	**0.592**	0.038	0.470	0.195	0.581	**0.962**	0.005	0.625	0.241	0.754	**0.999**	0.002	0.75	0.162	0.869
		(0.404)	(0.147)	(0.116)	(0.119)	(0.131)	(0.141)	(0.007)	(0.104)	(0.134)	(0.098)	(0.003)	(0.002)	(0.08)	(0.165)	(0.064)
$a = 1.8$	ERR	**0.174**	0.461	0.182	0.459	0.461	**0.036**	0.469	0.132	0.440	0.093	**0.005**	0.482	0.105	0.470	0.070
		(0.192)	(0.028)	(0.057)	(0.029)	(0.030)	(0.119)	(0.023)	(0.046)	(0.069)	(0.036)	(0.049)	(0.015)	(0.030)	(0.02)	(0.027)
	NMI	**0.524**	0.007	0.349	0.007	0.007	**0.908**	0.004	0.464	0.036	0.581	**0.989**	0.002	0.549	0.004	0.673
		(0.375)	(0.009)	(0.115)	(0.009)	(0.010)	(0.251)	(0.007)	(0.118)	(0.088)	(0.113)	(0.100)	(0.002)	(0.090)	(0.005)	(0.090)

Table 2: Results of simulation I: $K = 3$. The average misclassification rates ERR and normalized mutual information (NMI) are shown together with their standard deviations enclosed in parentheses for varying a in $f = a\sin(-8z_{ij})$, and varying number of nodes m.

		$m = 100$					$m = 200$					$m = 400$				
		FASBML	SBML	SPEC	JCDC	CASC	FASBML	SBML	SPEC	JCDC	CASC	FASBML	SBML	SPEC	JCDC	CASC
$a = 0$	ERR	**0.262**	0.265	0.298	0.422	0.276	**0.073**	0.075	0.185	0.376	0.158	**0.011**	**0.011**	0.074	0.372	0.064
		(0.133)	(0.133)	(0.067)	(0.067)	(0.068)	(0.095)	(0.096)	(0.045)	(0.032)	(0.032)	(0.005)	(0.005)	(0.021)	(0.023)	(0.013)
	NMI	**0.546**	0.544	0.404	0.238	0.435	**0.825**	0.824	0.545	0.346	0.588	**0.954**	0.953	0.753	0.398	0.783
		(0.110)	(0.110)	(0.077)	(0.089)	(0.086)	(0.060)	(0.063)	(0.060)	(0.043)	(0.062)	(0.020)	(0.020)	(0.045)	(0.024)	(0.036)
$a = 1.4$	ERR	0.380	0.535	0.407	0.499	**0.378**	**0.167**	0.524	0.352	0.477	0.316	**0.038**	0.534	0.335	0.449	0.287
		(0.098)	(0.052)	(0.065)	(0.073)	(0.059)	(0.149)	(0.041)	(0.057)	(0.090)	(0.054)	(0.089)	(0.037)	(0.055)	(0.107)	(0.047)
	NMI	**0.332**	0.099	0.272	0.125	0.295	**0.682**	0.117	0.331	0.179	0.364	**0.910**	0.117	0.351	0.257	0.400
		(0.142)	(0.071)	(0.072)	(0.089)	(0.075)	(0.176)	(0.057)	(0.056)	(0.126)	(0.062)	(0.076)	(0.056)	(0.050)	(0.166)	(0.058)
$a = 1.8$	ERR	**0.421**	0.566	0.450	0.566	0.432	**0.197**	0.573	0.434	0.590	0.401	**0.020**	0.592	0.436	0.619	0.387
		(0.094)	(0.044)	(0.059)	(0.055)	(0.057)	(0.154)	(0.040)	(0.059)	(0.051)	(0.061)	(0.008)	(0.030)	(0.053)	(0.049)	(0.059)
	NMI	**0.260**	0.053	0.209	0.054	0.221	**0.625**	0.050	0.244	0.035	0.268	**0.919**	0.038	0.270	0.020	0.300
		(0.125)	(0.049)	(0.068)	(0.058)	(0.071)	(0.195)	(0.037	(0.056)	(0.062)	(0.064)	(0.025)	(0.033)	(0.038)	(0.064)	(0.049)

as well as the shape of f are close to the true values. For example, in the case of f_1, $m = 200$ and $K = 3$, the average $\hat{\beta} = (0.1999, 0.9798)$ over 100 simulations, with standard deviations $(0.0099, 0.002)$. For f_2, $m = 200$ and $K = 3$, the average $\hat{\beta} = (-0.0006, 0.9998)$, with standard deviations $(0.0219, 0.0003)$. The community detection results are shown in Table **??**. We can see that the model tends to ignore the covariate that has no impact on the network, and the community

Table 3: Results of simulation II. The average misclassification rates (ERR) and normalized mutual information (NMI) are shown together with their standard deviations enclosed in parentheses for varying a in $f = a\sin(-\frac{4}{3}z_{ij})$, and varying number of nodes m. Numbers in bold indicate the best performance.

		$m=100$					$m=200$					$m=400$				
		FASBML	SBML	SPEC	JCDC	CASC	FASBML	SBML	SPEC	JCDC	CASC	FASBML	SBML	SPEC	JCDC	CASC
$a=0$	ERR	0.017	**0.011**	0.043	0.232	0.023	0.0008	**0.0005**	0.008	0.223	0.002	**0.000**	**0.000**	0.0002	0.227	**0.000**
		(0.016)	(0.010)	(0.028)	(0.041)	(0.020)	(0.002)	(0.002)	(0.007)	(0.025)	(0.003)	(0.0003)	(0)	(0.0006)	(0.017)	(0.000)
	NMI	0.894	**0.926**	0.773	0.333	0.863	0.994	**0.996**	0.945	0.382	0.985	**1.000**	**1.000**	0.998	0.378	**1.000**
		(0.087)	(0.065)	(0.117)	(0.077)	(0.101)	(0.016)	(0.012)	(0.049)	(0.037)	(0.024)	(0.002)	(0)	(0.006)	(0.023)	(0.002)
$a=1.4$	ERR	**0.041**	0.221	0.186	0.401	0.184	**0.004**	0.189	0.152	0.399	0.149	**0.0001**	0.189	0.132	0.415	0.138
		(0.027)	(0.064)	(0.041)	(0.056)	(0.045)	(0.004)	(0.053)	(0.028)	(0.062)	(0.030)	(0.0006)	(0.049)	(0.021)	(0.050)	(0.025)
	NMI	**0.778**	0.287	0.332	0.047	0.342	**0.968**	0.352	0.415	0.050	0.438	**0.999**	0.368	0.491	0.035	0.499
		(0.123)	(0.104)	(0.085)	(0.046)	(0.094)	(0.031)	(0.109)	(0.076)	(0.057)	(0.079)	(0.005)	(0.119)	(0.053)	(0.049)	(0.060)
$a=1.8$	ERR	**0.060**	0.256	0.231	0.440	0.235	**0.007**	0.251	0.220	0.436	0.226	**0.0002**	0.263	0.214	0.437	0.215
		(0.057)	(0.063)	(0.038)	(0.042)	(0.048)	(0.007)	(0.044)	(0.034)	(0.040)	(0.038)	(0.0007)	(0.036)	(0.034)	(0.034)	(0.034)
	NMI	**0.714**	0.226	0.236	0.017	0.236	**0.948**	0.232	0.257	0.018	0.257	**0.998**	0.216	0.272	0.016	0.290
		(0.159)	(0.094)	(0.075)	(0.021)	(0.093)	(0.048)	(0.070)	(0.069)	(0.020)	(0.079)	(0.007)	(0.056)	(0.073)	(0.014)	(0.077)

detection results are satisfactory in all cases and robust to nuisance covariates.

Table 4: Results of Simulation III. The average misclassification rates (ERR) and normalized mutual information (NMI) are shown for FASBML together with their standard deviations enclosed in parentheses for $f_1 = 1.8\sin(-\frac{4}{3}(0.2z_{1ij} + 0.9798z_{2ij}))$ and $f_2 = 1.8\sin(-\frac{4}{3}(z_{2ij}))$, with varying number of nodes m.

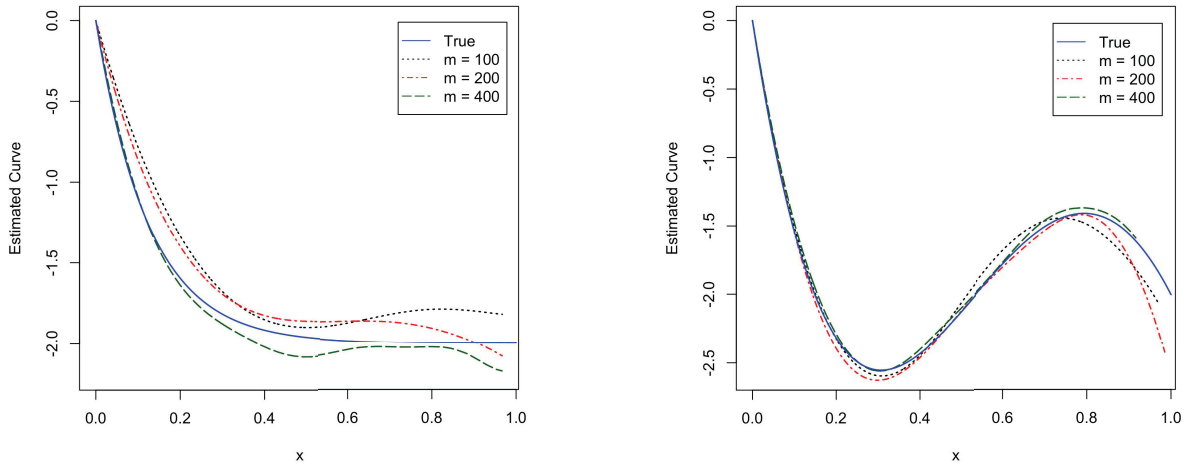
		$m=100$		$m=200$		$m=400$	
		f_1	f_2	f_1	f_2	f_1	f_2
$K=2$	ERR	0.041	0.073	0.002	0.011	0.0001	0.0004
		(0.031)	(0.048)	(0.004)	(0.009)	(0.0004)	(0.0009)
	NMI	0.782	0.659	0.981	0.923	0.9995	0.996
		(0.146)	(0.163)	(0.027)	(0.054)	(0.003)	(0.009)
$K=3$	ERR	0.259	0.292	0.051	0.049	0.010	0.010
		(0.124)	(0.126)	(0.034)	(0.015)	(0.005)	(0.005)
	NMI	0.460	0.429	0.835	0.834	0.956	0.956
		(0.161)	(0.163)	(0.058)	(0.043)	(0.020)	(0.019)

In **Simulation IV**, we use two more examples to illustrate the empirical performance of the nonparametric estimation for the function f. We set $K = 2$ in both examples. In the first example, f is an exponential function, $f(z_{ij}) = 2\exp(-8z_{ij}) - 2$; in the second example, f is a polynomial

Table 5: Results of Simulation IV. The average misclassification rates (ERR) and normalized mutual information (NMI) are shown for FASBML together with their standard deviations enclosed in parentheses for exponential f and polynomial f, with varying number of nodes m.

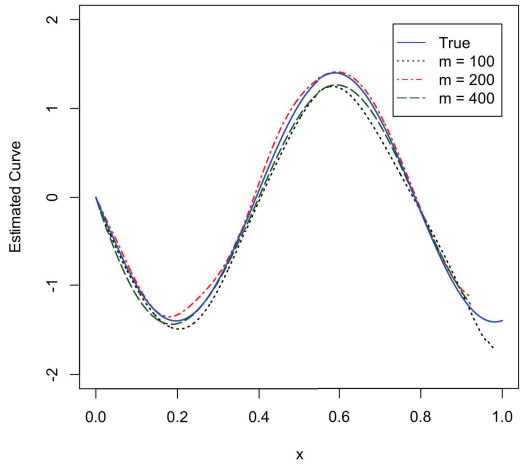
	$m = 100$		$m = 200$		$m = 400$	
	Exp f	Poly f	Exp f	Poly f	Exp f	Poly f
ERR	0.098	0.172	0.021	0.046	0.002	0.006
	(0.056)	(0.090)	(0.012)	(0.040)	(0.003)	(0.004)
NMI	0.574	0.398	0.866	0.763	0.981	0.954
	(0.136)	(0.178)	(0.067)	(0.130)	(0.021)	(0.029)

function, $f(z_{ij}) = 10z_{ij}^4 - 42z_{ij}^3 + 50z_{ij}^2 - 20z_{ij}$. A fitted curve randomly selected from 100 simulations is depicted in Figure 1 for each scenario. It is shown that, when the network is of moderate size, the fitted curve is remarkably close to the true curve, except some boundary effect near the endpoints. The proposed algorithm can also provide satisfying partition results as presented in Table 5.

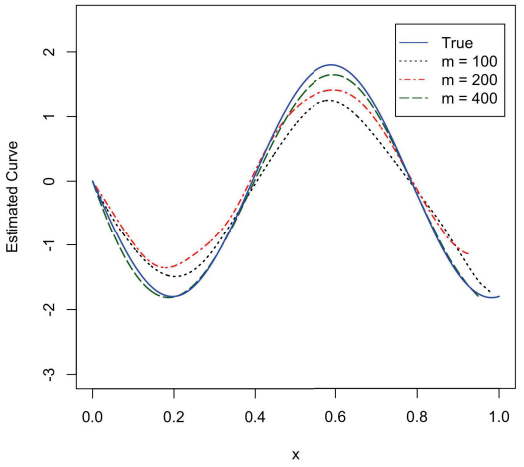


(a)

(b)



(c)



(d)

Figure 1: Estimates of f for a randomly selected simulated network with varying f functions and varying number of nodes m. (a) $f(x) = 2\exp(-8x) - 2$. (b) $f(x) = 10x^4 - 42x^3 + 50x^2 - 20x$. (c) $f(x) = 1.4sin(-8x)$. (d) $f(x) = 1.8sin(-8x)$

.

5 Data Applications

In this section, we show applications of our method to two actual world networks: a functional brain network and an US air-transportation network, which are representative examples of biological and infrastructure systems. The proposed FASBM reveal interesting node feature effects, as well as interpretable communities.

5.1 Functional Brain Network

We first consider an application to brain functional connectivity study using resting-state functional magnetic resonance imaging (RS-fMRI) data. The data were collected by University of Pittsburgh Medical Center and detailed descriptions can be found in Hwang et al. (2013). Imaging data were preprocessed to reduce noise and artifacts using standard fMRI data processing methods .

RS-fMRI measures the intrinsic, high-amplitude, low-frequency blood-oxygen-level dependence signal (BOLD) fluctuations of the brain. The relationship between RS-fMRI signals from different regions is thought to reflect functional connectivity independent of any particular brain state (Van Dijk et al., 2010). Functional connectivity between a pair of voxels is usually estimated by calculating the Pearson correlation coefficient between their BOLD time series, treating the observations as coming from a single bivariate distribution.

The brain network in this analysis contains 448 nodes (voxels) in the basal ganglia mask. The data matrix Y_{ij} is the averaged Fisher's z-transformed correlation values based upon all subjects. The basal ganglia subserves a wide range of functions, including motor, cognitive, motivational, and emotional processes and has been implicated in numerous neurological and psychiatric disorders. There have been great interest in using RS-fMRI techniques to study the functional connectivity in basal ganglia (Di Martino et al., 2008; Robinson et al., 2009; Barnes et al., 2010).

Given the fact that connectivity between adjacent nodes is sometimes over-represented due to inevitable technical reasons in fMRI data acquisition process and data processing (Stanley et al., 2013), we consider the Euclidean distance between two voxels as the covariate z_{ij} in applying

FASBM to discover the underlying block structure of the functional brain network. Here the spatial location of each node is defined as the coordinates of the center of the voxel in MNI stereotactic space.

The estimated function f as shown in Figure 2d reflect the expected relationship between brain connectivity and spatial locations. Figure 2e reveals that the pairs of voxels within the same block are not exactly connected in the same way as evidenced by the noise patterns within blocks. Fitting of the simple stochastic block model to the brain network can not characterize the heterogeneity within blocks, whereas the proposed FASBM with spatial feature z_{ij} incorporated is a better approximation to the data by accounting for the spurious connection between adjacent nodes. As shown in Figure 2f: the nonparametric function f in our model captures the additive effect of the deviations from the block structure. It can be seen that the heterogeneity within the blocks are well explained by the effect of local correlations as modeled by the nonparametric function f.

As shown in the top panels of Figure 2, using FASBML yields functionally distinct but spatially coherent parcellations of the brain region. Previous studies have parcellated the basal ganglia based on its extrinsic functional connectivity with the cortex (Barnes et al., 2010; Choi et al., 2012). It is unknown that whether or not the basal ganglia can be successfully parcellated by only considering local, intrinsic functional information within the basal ganglia. Using the proposed method, we have successfully identified basal ganglia subdivisions by only considering functional connectivity pattern between basal ganglia voxels. This parcellation closely resembled those reported using structural anatomical information (Tziortzi et al., 2011). By visual examination: cluster 1(yellow) corresponds to the caudate body, cluster 3(green) corresponds closely to the putamen, cluster 5(cyan) closely to the pallidum , and cluster 2(red), 4(blue) partially correspond to the caudate head.

5.2 United States Air-transportation Network

For the second example, we analyze a US airline network. We extracted information of the United States domestic airports and flights for the year 2012 from the OpenFlights/Airline Route Mapper Route Database. The resulting air-transportation network comprises 300 nodes denoting

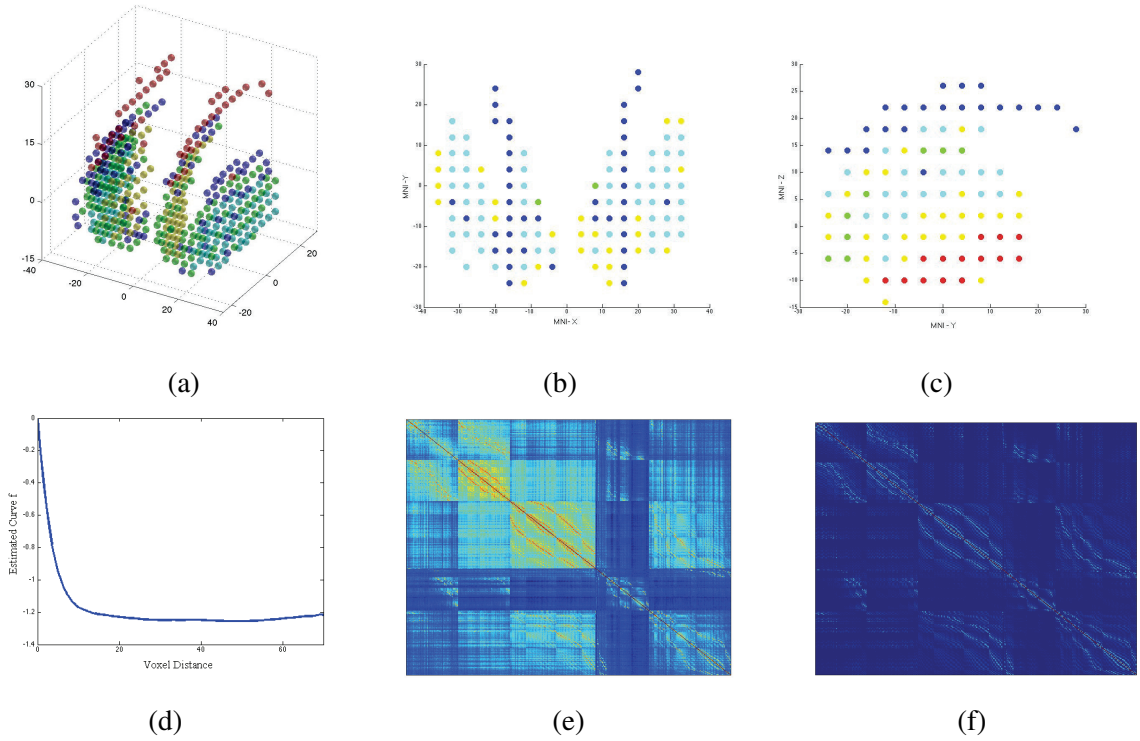


Figure 2: (a) The functional brain network: each voxel was represented by a single node at its spacial location with the color reflecting the inferred community membership by the proposed FASBML. (b) Projection of (a) in the x-y plane of the Montreal Neurological Institute (MNI) stereotactic space. (c) Projection of (a) in the MNI y-z plane. (d) Estimated f function. (e) Connectivity matrix of the brain network data with voxels ordered by inferred community membership. (f) Fitted f evaluated on the distance matrix z_{ij} of the brain network data with voxels ordered by inferred community membership.

airports in the United States and about 6000 flight routes within the United States operated by the major airlines (United Airlines (UA), American Airlines(AA), Delta Air Lines and Southwest Airlines). The edges in the network indicate presence or absence of non-stop flights between two airports. The full data set can be downloaded from `http://www.openflights.org`.

The air-transportation network is a complex network with heterogeneous degrees: a handful of nodes in the air transportation network are busy airports having a significant number of connections to and from other airports. Therefore, it is expected that community-detection methods solely based on the adjacency matrix will tend to form communities characterized by different

degrees. For instance, SBML split the network into four groups by degree: high, relatively high, medium and low.

In the following, we fit the proposed feature adjusted stochastic block model (FASBM) in the hope to discover community structures that are not merely due to the degree distribution. The node feature we consider is the number of airports it has connections to, i.e., $f_i = \sum_{l=1}^{m} Y_{il}$, and let $z_{ij} = f_i + f_j$. The use of FASBM requires a pre-specified number of communities as input, whereas it is unclear how many communities are in the airline network, 2-fold network cross-validation (NCV) was applied to determine the number of communities. The NCV approach is recently proposed by Chen and Lei (2018) to select the number of clusters through block-wise edge splitting. With the negative log-likelihood as the loss functions, the NCV method consistently selects $K = 4$ communities.

The community labeled in orange identifies almost all the "home base" airports of Southwest airline: *Las Vegas McCarran Int'l, Houston Hobby Int'l, Chicago Midway Int'l, Baltimore-Washington Int'l, Lambert-St. Louis Int'l, Nashville Int'l and Kansas City Int'l, Austin-Bergstrom Int'l* and so on. The community labeled in red mainly consists of airports served as hubs for UA, AA or Delta airlines, including *Hartsfield Jackson Atlanta Int'l* and *Detroit Metropolitan Airport* as hubs for Delta, *Chicago O'hare Int'l*, *Newark Liberty Int'l* and *Washington Dulles Int'l* as hubs for UA, *Philadelphia Int'l*, *Charlotte Douglas Int'l* and *Ronald Reagan Washington National Airport* as hubs for AA. The community labeled in green comprises airports characterized by varying node degrees, where the low degree airports have one of UA, AA or Delta airlines as the only carrier, and busy airports serve as hubs for one of the UA, AA and Delta airlines. The community labeled in blue corresponds to airports with low degree. Many members of this community are regional airports that serve air traffic within a relatively local or lightly populated regions. Additionally, we have shown in Figure 4 that, the shape of the estimated f function reflects the general relationship between connectivity probability and the sum of degrees for a pair of nodes - airports with high degrees tend to connect to other airports, and the opposite holds true for low degrees airports.

Our results are in agreement with the fact that Southwest, as the fourth largest airlines in the U.S., after the big three legacy carriers (UA, AA and Delta), was less assertive in big travel markets and

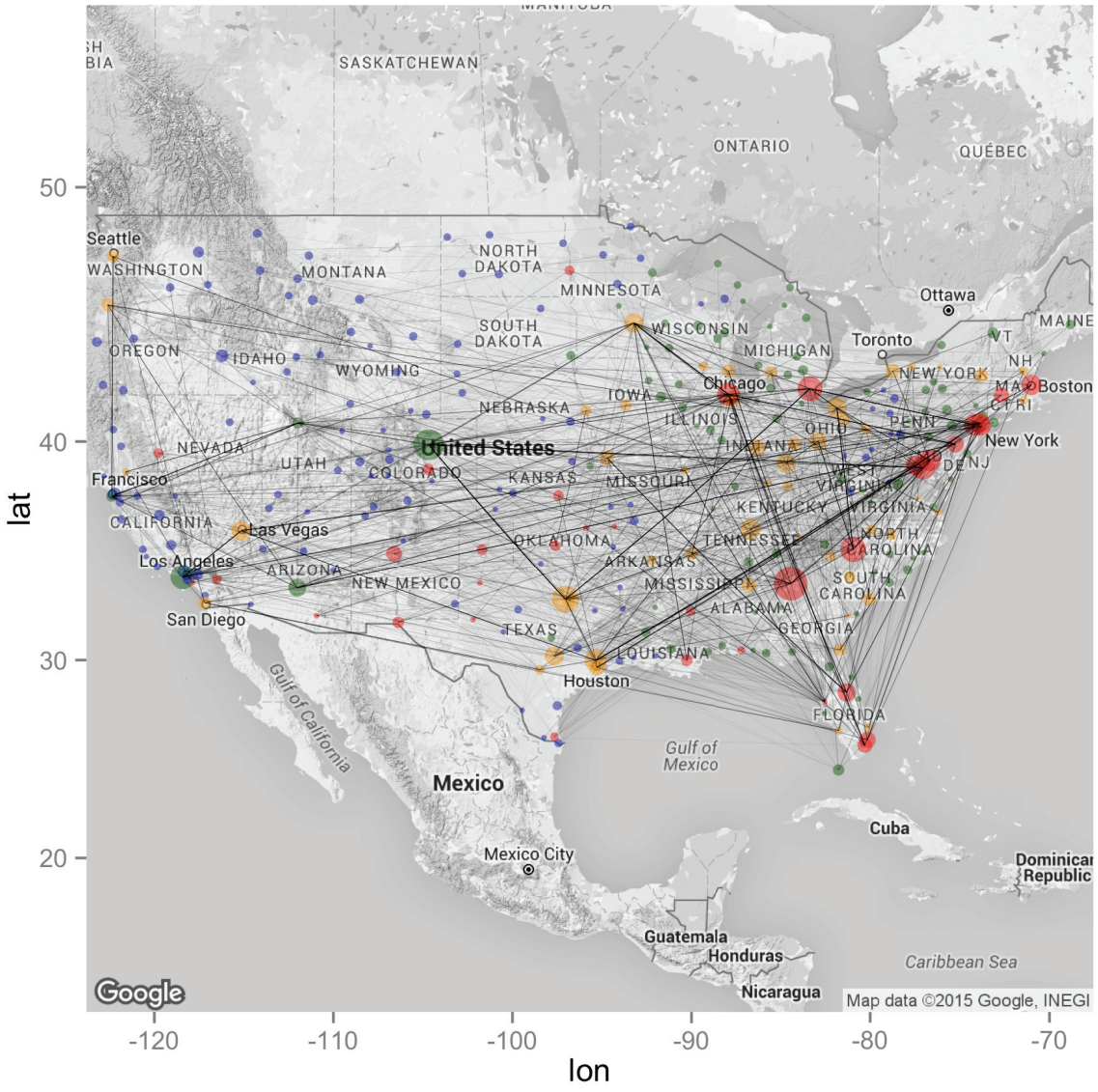


Figure 3: The communities inferred by Feature adjusted stochastic block model (FASBM). The size of the vertex is proportional to the square root of the node degree and the color reflects inferred community membership: Orange labels the community corresponding to almost all the home base airports of Southwest airline; red labels the community of hubs for UA, AA or Delta; green labels the community comprising airports characterized by varying node degrees; blue labels the community of regional airports.

chose to avoid competing with the "big three" in their hub airports, and instead focuses on cities other than these big hubs. Southwest Airlines adopts a point-to-point (PP) configuration wherein airports are connected by direct routes. On the contrary, the "big three" adopt the hub-and-spoke

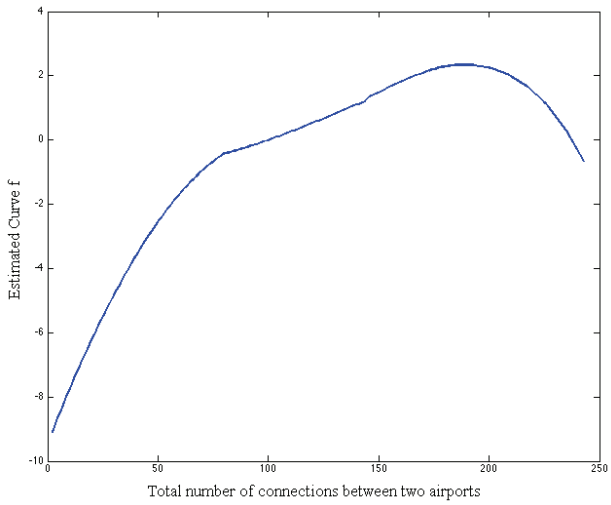


Figure 4: Estimated f function for the US air-transportation network.

(HS) system (Aguirregabiria and Ho, 2010) , wherein most of the operations are concentrated in the hubs and all other cities in the network (i.e., the spokes) are connected to the hubs by non-stop flights. Although there is no "correct" way to partition the air transportation network, compared to the partition by SBM, FASBM allows us to recover the hidden structural organization that is beyond the groups of degrees. The node feature information incorporated in the block model helps to provide more insights into the development and categorization of the air-transportation network.

6 Discussion

In this paper, we have demonstrated how one can incorporate node feature information upon stochastic block models, focusing on the problem of community detection beyond that explained by the node features as well as learning the influence of features on the network topology. The empirical results show that the proposed method can estimate f non-parametrically, requiring no prior knowledge of how and the extent to which the network is affected by the features. The proposed feature adjusted stochastic block model (FASBM) can be used as a generative model for estimation and prediction in networks, making probabilistic statements about the impact of features and so on. Useful extensions include models for directed networks and overlapped com-

munities, and we leave these for future work.

In the following, we discuss several computational issues. First, the local polynomial maximum likelihood estimation of f sometimes called be computational intensive. To produce Table 3 in simulation II, for the case $m = 400$, $a = 1.8$, and 100 simulation repetitions, our algorithms fitting FASBM runs 1.5 hours on a Macbook Pro with 8GB RAM processor and 2 GHz Intel Core i7. Fan and Chen (1999) and Cai et al. (2000) proposed to replace the iterative local MLE with the one-step Newton Ralphson estimator and proved in theory that the one-step local MLE does not deteriorate performance as long as the initial estimator is reasonably accurate. The choice of bandwidth in the estimation of f controls how smooth the fit is. Since we have $m \times (m-1)/2$ data points for the curve fitting, the design is very dense. Our practical experience suggests that use of one-tenth of the total range as bandwidth usually results in a relatively smooth f function. Other data-driven methods developed in kernel smoothing although time consuming can also be used. Given that the design can be extremely dense and the curve is usually fairly smooth, we implemented the option allowing one to randomly sample a grid of points to fit the curve. Alternative methods such as binned and updated method (Fan and Marron, 1994) can also be considered. In addition to these accelerating methods, one can also adopt other non-parametric smoothing methods to estimate f. We choose local polynomial approximation mainly because it produce the estimate of derivative function at almost no additional cost. Second, like the classic SBM and its variants, the number of communities K in the FASBM has to be pre-specified. In the paper we adapted the network cross-validation (NCV) method for the stochastic block model proposed by Chen and Lei (2018), because the extension of NCV to FASBM is conceptually straightforward. Recently, there are other methods of choosing K developed for the SBM based on likelihood approaches, which might also be useful for FASBM. Last but not least, we used greedy-algorithm to avoid a full search of the possible partitions in the model fitting. This algorithm works very well in practice but so far there is no theoretical guarantee of the convergence to the global maximum. We believe that the development of approximation theories for these greedy algorithms is of interest.

References

Victor Aguirregabiria and Chun-Yu Ho. A dynamic game of airline network competition: Hub-and-spoke networks and entry deterrence. *International Journal of Industrial Organization*, 28(4):377–382, 2010.

Kelly Anne Barnes, Alexander L Cohen, Jonathan D Power, Steven M Nelson, Yannic BL Dosenbach, Francis M Miezin, Steven E Petersen, and Bradley L Schlaggar. Identifying basal ganglia divisions in individuals using resting-state functional connectivity mri. *Frontiers in systems neuroscience*, 4, 2010.

Peter J Bickel and Aiyou Chen. A nonparametric view of network models and newman–girvan and other modularities. *Proceedings of the National Academy of Sciences*, 106(50):21068–21073, 2009.

Peter J Bickel and Purnamrita Sarkar. Hypothesis testing for automated community detection in networks. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2015.

N. Binkiewicz, J. T. Vogelstein, and K. Rohe. Covariate-assisted spectral clustering. *Biometrika*, 104(2), 2017.

Zongwu Cai, Jianqing Fan, and Runze Li. Efficient estimation and inferences for varying-coefficient models. *Journal of the American Statistical Association*, 95(451):888–902, 2000.

Raymond J Carroll, Jianqing Fan, Irene Gijbels, and Matt P Wand. Generalized partially linear single-index models. *Journal of the American Statistical Association*, 92(438):477–489, 1997.

Alain Celisse, Jean-Jacques Daudin, Laurent Pierre, et al. Consistency of maximum-likelihood and variational estimators in the stochastic block model. *Electronic Journal of Statistics*, 6: 1847–1899, 2012.

Kehui Chen and Jing Lei. Network cross-validation for determining the number of communities in network data. *Journal of the American Statistical Association*, 113(521):241–251, 2018.

David S Choi, Patrick J Wolfe, and Edoardo M Airoldi. Stochastic blockmodels with a growing number of classes. *Biometrika*, page asr053, 2012.

A Di Martino, A Scheres, DS Margulies, AMC Kelly, LQ Uddin, Z Shehzad, B Biswal, JR Walters, FX Castellanos, and MP Milham. Functional connectivity of human striatum: a resting state fmri study. *Cerebral cortex*, 18(12):2735–2747, 2008.

J. Fan and I. Gijbels. *Local Polynomial Modelling and its Applications*. Chapman & Hall, London, 1996. ISBN 0-412-98321-4.

Jianqing Fan and Jianwei Chen. One-step local quasi-likelihood estimation. *Journal of the Royal Statistical Society. Series B, Statistical Methodology*, pages 927–943, 1999.

Jianqing Fan and James S Marron. Fast implementations of nonparametric curve estimators. *Journal of Computational and Graphical Statistics*, 3(1):35–56, 1994.

Michelle Girvan and Mark Newman. Community structure in social and biological networks. *Proceedings of the national academy of sciences*, 99(12):7821–7826, 2002.

Anna Goldenberg, Alice Zheng, Stephen Fienberg, and Edoardo Airoldi. A survey of statistical network models. *Foundations and Trends® in Machine Learning*, 2(2):129–233, 2010.

Mark S Handcock, Adrian E Raftery, and Jeremy M Tantrum. Model-based clustering for social networks. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 170(2): 301–354, 2007.

Peter D Hoff, Adrian E Raftery, and Mark S Handcock. Latent space approaches to social network analysis. *Journal of the american Statistical association*, 97(460):1090–1098, 2002.

Paul W Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. *Social networks*, 5(2):109–137, 1983.

K. Hwang, M.N. Hallquist, and B. Luna. The development of hub architecture in the human functional brain network. *Cerebral Cortex*, 23(10):2380–2393, 2013.

Brian Karrer and Mark Newman. Stochastic blockmodels and community structure in networks. *Physical Review E*, 83(1):016107, 2011.

Eric D Kolaczyk. *Statistical Analysis of Network Data: Methods and Models*. Springer Science & Business Media, 2009.

Tarald O Kvalseth. Entropy and correlation: Some comments. *IEEE Transactions on Systems, Man, and Cybernetics*, 3(17):517–519, 1987.

Jing Lei, Alessandro Rinaldo, et al. Consistency of spectral clustering in stochastic block models. *The Annals of Statistics*, 43(1):215–237, 2014.

Xin Liu, Tsuyoshi Murata, and Ken Wakita. Detecting network communities beyond assortativity-related attributes. *Physical Review E*, 90(1):012806, 2014.

Miller McPherson, Lynn Smith-Lovin, and James M Cook. Birds of a feather: Homophily in social networks. *Annual review of sociology*, pages 415–444, 2001.

Mark EJ Newman and Aaron Clauset. Structure and inference in annotated networks. *Nature Communications*, 7, 2016.

Mark EJ Newman and Michelle Girvan. Finding and evaluating community structure in networks. *Physical review E*, 69(2):026113, 2004.

Krzysztof Nowicki and Tom Snijders. Estimation and prediction for stochastic blockstructures. *Journal of the American Statistical Association*, 96(455):1077–1087, 2001.

Simon Robinson, Gianpaolo Basso, Nicola Soldati, Uta Sailer, Jorge Jovicich, Lorenzo Bruzzone, Ilse Kryspin-Exner, Herbert Bauer, and Ewald Moser. A resting state network in the motor control circuit of the basal ganglia. *BMC neuroscience*, 10(1):137, 2009.

Karl Rohe, Sourav Chatterjee, and Bin Yu. Spectral clustering and the high-dimensional stochastic blockmodel. *The Annals of Statistics*, 39:1878–1915, 2011.

Tom Snijders and Krzysztof Nowicki. Estimation and prediction for stochastic blockmodels for graphs with latent block structure. *Journal of Classification*, 14(1):75–100, 1997.

Matthew L Stanley, Malaak N Moussa, Brielle M Paolini, Robert G Lyday, Jonathan H Burdette, and Paul J Laurienti. Defining nodes in complex brain networks. *Frontiers in computational neuroscience*, 7, 2013.

Matthew Stephens. Dealing with label switching in mixture models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 62(4):795–809, 2000.

Andri C Tziortzi, Graham E Searle, Sofia Tzimopoulou, Cristian Salinas, John D Beaver, Mark Jenkinson, Marc Laruelle, Eugenii A Rabiner, and Roger N Gunn. Imaging dopamine receptors in humans with [11 c]-(+)-phno: dissection of d3 signal and anatomy. *Neuroimage*, 54(1):264–277, 2011.

Koene RA Van Dijk, Trey Hedden, Archana Venkataraman, Karleyton C Evans, Sara W Lazar, and Randy L Buckner. Intrinsic functional connectivity as a tool for human connectomics: theory, properties, and optimization. *Journal of neurophysiology*, 103(1):297–321, 2010.

Emmanuel Viennet et al. Community detection based on structural and attribute similarities. In *ICDS 2012, The Sixth International Conference on Digital Society*, pages 7–12, 2012.

Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

Jaewon Yang, Julian McAuley, and Jure Leskovec. Community detection in networks with node attributes. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, pages 1151–1156. IEEE, 2013.

Yuan Zhang, Elizaveta Levina, Ji Zhu, et al. Community detection in networks with node features. *Electronic Journal of Statistics*, 10(2):3153–3178, 2016.

Yunpeng Zhao, Elizaveta Levina, Ji Zhu, et al. Consistency of community detection in networks under degree-corrected stochastic block models. *The Annals of Statistics*, 40(4):2266–2292, 2012.